Computational Complexity, Protein Structure Prediction, and the Levinthal Paradox

J. Thomas Ngo, Joe Marks, and Martin Karplus

1. Perspectives and Overview

A protein molecule is a covalent chain of amino acid residues. Although it is topologically linear, in physiological conditions it folds into a unique (though flexible) three-dimensional structure. This structure, which has been determined by x-ray crystallography and nuclear magnetic resonance for many proteins (Bernstein et al., 1977; Abola et al., 1987), is referred to as the native structure. As demonstrated by the experiments of Anfinsen and co-workers (Anfinsen et al., 1961; Anfinsen, 1973), at least some protein molecules, when denatured (unfolded) by disrupting conditions in their environment (such as acidity or high temperature) can spontaneously refold to their native structures when proper physiological conditions are restored. Thus, all of the information necessary to determine the native structure can be contained in the amino acid sequence.

From this observation, it is reasonable to suppose that the native fold of a protein can be predicted *computationally* using information only about its chemical composition. In particular, it should be possible to write down a mathematical problem that, when solved, gives the native conformation of the protein. This procedure would be self-contained, in the sense that no additional information about the biology of protein synthesis would be required. Further, it is reasonable to hope that this procedure could be accomplished without requiring an astronomical amount of computer resources, given the observation that polypeptide chains do fold to their

The Protein Folding Problem and Tertiary Structure Prediction
K. Merz, Jr. and S. Le Grand, Editors
© Birkhäuser Boston 1994

which depends steeply on the energy gap U. Given the assumptions that $N = 100$ and $n = 2$, it was found that in the limit $U \rightarrow 0$, the first-passage time is nearly 10^{30} years. However, a modest change to the value of U, say $U = 2kT$, lowers the first-passage time to under one second. (The base of the exponential, $1 + n\exp(-U/kT)$, is equal to 3 when $U = 0$, but 1.27 when $U = 2kT$.)

The analysis of Zwanzig et al. resolves a form of the Levinthal paradox in which the absence of clues about the form of the native state is the sole basis for expecting exponential-time folding. However, it does not resolve the form of the paradox based on computational complexity, since the optimization problem implied by the underlying model can be solved trivially in linear time. The reason for the tractability of the underlying model is the lack of long-range interactions, which are critical to rendering PSP NP-hard (Ngo and Marks, 1992), and essential for cooperativity (Karplus and Shakhnovich, 1992).

6. Future Work

It is not known whether there exists an efficient algorithm for predicting the structure of a given protein from its amino acid sequence alone. Decades of research have failed to produce such an algorithm, yet Nature seems to solve the problem. Proteins do fold! The "direct" approach to structure prediction, that of directly simulating the folding process, is not yet possible because contemporary hardware falls eight to nine orders of magnitude short of the task. However, while this difference is large, it is not astronomical. Would this "direct" approach constitute an efficient and correct algorithm for protein-structure prediction? Too little is known about protein folding, and about the future of computing technology, to be able to answer this question at this time.

The results reviewed here (Section 3) do not completely rule out the existence of a protein-structure prediction algorithm that is both efficient and correct, in the precise senses of those words used throughout this chapter. In particular, it remains formally possible that there is a restricted form of PSP that is efficiently solvable, but subsumes protein-structure prediction. How can this possibility be investigated?

A standard strategy in the analysis of any NP-hard problem is to examine restricted forms of the problem systematically, classifying each as tractable or NP-hard, and thereby exposing the sources of the complexity. Barahona's results with Ising spin-glass models, which were described briefly in Section 4, are exemplary of this approach. While the particular restrictions chosen by Barahona for spin glasses (reduction of dimensionality and removal of the magnetic field) are not suitable for protein-structure prediction, the overall strategy of examining restricted forms is appropriate. Some restricted form of PSP in which compactness plays a critical role is a candidate for this type of analysis (Section 4.6).

The approach of considering restricted forms has worked well for dozens of important problems that are relatively "clean" and abstract (Garey and Johnson, 1979), but it may be difficult to pursue in the case of protein-structure prediction. In the former case, the problem shown to be NP-hard is usually as general as would actually be required in practice. In the latter case, what is desired is not an algorithm that can handle all possible instances of PSP (Section 3), but merely one that works for proteins. Thus, the fact that PSP is a generalization of protein-structure prediction makes the result that PSP is NP-hard less limiting than it could be.

Ideally, one would like to demonstrate the NP-hardness of a problem that is more *specific*, not more general, than protein-structure prediction, because that would automatically prove the NP-hardness of protein-structure prediction itself. This would entail finding an efficient transformation from some existing NP-complete problem that generates instances of PSP that are proteins by every conceivable criterion.[38] It is difficult to see how such a transformation might proceed.[39]

An alternative approach that may be nearly as instructive is to use the currently available result regarding PSP as a baseline in a continuing comparative analysis—to find restricted forms of PSP that are NP-hard but as specialized as possible, and to find others that are tractable but as general as possible. The motivations for pursuing this methodology are both practical and theoretical:

- Every NP-hardness result permits us to know in advance that a certain group of algorithms is likely to fail, and is therefore not worth pursuing (Section 4).
- Conversely, every NP-hardness result helps identify a source of complexity in protein-structure prediction, and therefore what must be stripped away from the problem before it is reasonable to attempt efficient solution.

  The work of Finkelstein and Reva (1992) is a good example: an approach to structure prediction with a guaranteed polynomial time bound was developed. The critical assumption behind the algorithm is that only nonbonded interactions between nearest neighbors along the chain are significant. Because of this assumption, the algorithm cannot solve all instances of PSP, but instead is restricted to instances in which only nonbonded interactions between nearest neighbors along the chain are nonzero.[40] This violates the requirements of the reduction from Partition to PSP, in which nonbonded interactions between sites distant from each other along the chain are essential. Thus, the problem is similar in character to that examined by Zwanzig et al. (Section 5.3). While the Finkelstein–Reva algorithm was not inspired by an NP-hardness result, the underlying strategy is similar to how NP-hardness results might be used; they removed from the problem what they observed to be a source of complexity. However, in this case, removing the source of complexity led to a problem different from that posed by protein folding, in which long-range interactions play an essential role.

- The NP-hardness of PSP serves as the premise for a reformulation of the Levinthal paradox (Section 5), whose conventional form is based on a model of folding that is in conflict with known experimental results. A motivation for pursuing an analysis of the computational complexity of protein-structure prediction is to assist in the constructive role of the Levinthal paradox—to help focus attention on the key questions in protein folding.

  A small number of reasonably well-defined potential resolutions to the computational-complexity form of the Levinthal paradox were listed in Section 5. One of the possible resolutions is that protein-structure prediction is tractable. NP-hardness results with restricted forms of PSP would make that possible resolution less likely, thus lending credence to the alternatives.

  Attempts to resolve the Levinthal paradox, which play a valid and useful role in helping to understand how proteins fold, can lead to confusion because the premises of the original form of the paradox are not well formulated. In particular, one such proposed resolution (Zwanzig et al., 1992) can be shown unequivocally not to resolve the computational complexity form of the paradox, and in related arguments (Karplus and Shakhnovich, 1992) has been shown to lead to physically incorrect consequences (Section 5.3). For the paradox to be meaningful, it must be "falsifiable"—it must be possible to know when the paradox has been resolved.

In addition to restricted forms of PSP, it would be useful to know the computational complexity of other tasks in structure prediction that appear easier than the general problem, but whose complexities are none the less uncertain.

The task of computing side-chain conformations given full knowledge of a protein's backbone conformation is one such problem. Case studies using simulated annealing (Lee and Subbiah, 1991) have suggested that packing effects may suffice to determine, in part, the side-chain conformations in a protein's core. The computational complexity of this packing problem is unknown. Because only short-range effects are present, the graph of possible side-chain–side-chain interactions can be known in advance, is sparse, and consists of vertices of low degree. Previous experience—for instance, with Ising spin-glass models (Barahqna, 1982), graph colorability (Garey and Johnson, 1979, p. 191) and cartographic labeling (Formann and Wagner, 1991; Marks and Shieber, 1991)—illustrates that such neighborhood interactions can, on their own, give rise to NP-hardness. On the other hand, many problems that contain such neighborhood interactions are tractable if restrictions can be placed on the nature of the graph (Garey and Johnson, 1979), suggesting that the problem of finding a mutually acceptable set of side-chain conformations for a protein could be tractable. (One currently known algorithm for predicting side-chain conformations based on backbone positions achieves 70% to 80% accuracy for χ_1 and χ_2 angles (Dunbrack and Karplus, 1993).) Not knowing the computational complexity of side-chain structure prediction leaves the algorithm developer in the quandary of not knowing whether inexact methods are truly necessary, given the possible existence of a superior exact algorithm.

Acknowledgments. We thank Ron Unger for answering detailed questions and providing a preprint (Unger and Moult, 1993). Aviezri Fraenkel also kindly provided a preprint (Fraenkel, 1993). We thank Harry Lewis, Eugene Shakhnovich, and Jim Clark for reading and commenting on the manuscript. JTN is grateful for a Graduate Fellowship from the Fannie and John Hertz Foundation. This research was supported in part by grants from the National Science Foundation and the National Institutes of Health.

NOTES

[1] The Thermodynamic Hypothesis states that a protein's native fold is the configuration of globally minimal free energy. However, it is generally assumed that a protein's states of lowest free energy are similar enough in entropy to justify the use of potential energies instead of free energies as a computational convenience: potential energies are much faster and more straightforward to compute.

[2] For example, if only nonbonded interactions between nearest neighbors along the chain are significant, the global minimum structure can be predicted efficiently (Finkelstein and Reva, 1992).

[3] The term *combinatorial optimization* is normally reserved for problems in which the solution space is discrete. Throughout this chapter we use the term to refer